

# ISAE, Domaine "Systèmes autonomes"

## BE Planification de tâches (en binôme)

Date du BE: 19/10/2018

### Important:

Le BE est à faire entièrement. Il n'est pas à rendre, mais **ses résultats seront nécessaires pour l'examen individuel noté du 26/10/2018**. Pour les élèves ayant des difficultés à terminer le BE avant l'examen du 26/10, une séance supplémentaire d'assistance pourra être mise en oeuvre. Chaque élève doit être capable de lancer les planificateurs étudiés (ff et optic-clp), de modifier les modèles (domaines et problèmes), d'analyser les résultats, et de copier/coller les modèles et les résultats dans l'interface d'examen. Il est demandé à chaque étudiant de conserver dans son espace personnel le travail du binôme: i.e. les résultats de ce BE en vue de leur consultation/modification lors de l'examen individuel.

### Sujet : Robots agricoles

Une ferme maraîchère s'est équipée de plusieurs robots de type robot terrestre (Fig 1.a) qui peuvent porter divers outils pour le buttage (brosses Fig 1.b) et le binage (socle Fig 1.c)



Figure 1.a



Figure1.b



Figure1.c

Les tâches que doivent effectuer ces robots sont le buttage et le binage. Dans ce problème, un ou plusieurs robots, équipés des instruments adéquats pour chaque action, aident à la réalisation des tâches. On se propose d'étudier deux logiciels de planification globale pour trouver les plans d'actions des robots.

Les hypothèses simplificatrices pour ce BE sont les suivantes:

- l'environnement est décrit par un graphe de zones interconnectées à partir d'une base (par exemple, les graphes en figures 2.a et 2.b);
- 2 types de tâches sont considérées : buttage et binage;
- 2 types d'instruments amovibles sont utilisés pour chaque tâche : le socle de binage et les brosses de buttage (il faut alors choisir quel outil installer).

Les robots peuvent effectuer les actions suivantes:

- **move**: se déplacer entre deux lieux *adjacents*;
- **mount** : s'équiper d'un instrument (socle de binage, ou brosses de buttage), s'il n'en porte pas et s'il est à la base ;
- **unmount** : enlever l'instrument qu'il porte, s'il est à la base;
- **perform** : effectuer une des deux tâches dans une zone, s'il s'y trouve et s'il porte l'instrument adapté.

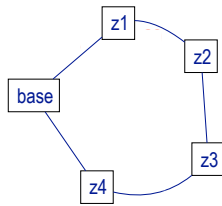


Figure 2.a

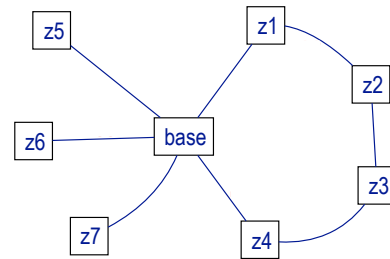


Figure2.b

Le BE porte sur la modélisation de quelques problèmes et leur résolution en utilisant le langage PDDL avec les planificateurs **ff** (open source <https://fai.cs.uni-saarland.de/hoffmann/ff.html>) et le planificateur **optic** (open source <https://nms.kcl.ac.uk/planning/software/optic.html>) tous les deux disponibles sur les ordinateurs de l'ISAE. L'annexe de ce document explique comment utiliser ces planificateurs. Pour se concentrer sur les points intéressants, on fournit pour chaque question :

- deux fichiers problèmes pbXa.pddl et pbXb.pddl contenant les spécifications complètes de 2 instances du problème correspondant aux deux figures 2.a et 2.b;
- un fichier avec une définition partielle en PDDL domX.pddl (pour ff) ou domXd.pddl (d pour duration, pour optic) à compléter avec la spécification des actions pour résoudre la question.

Pour les questions impaires 1, 3, 5, on utilisera le planificateur ff qui utilise des modèles d'action PDDL "classiques". Pour les questions paires, 2, 4, 6 on utilise le planificateur optic qui utilise des modèles d'action "durative actions", prenant en compte la durée réelle des actions.

On dispose d'un seul robot terrestre.

**Question 1 (ff):** Spécifier dans **dom1.pddl** les quatre actions nécessaires à la planification des tâches décrites dans les fichiers **pb1a.pddl** et **pb1b.pddl**. Commenter les plans produits.

**Question 2 (optic):** Spécifier dans **dom1d.pddl** (d pour durative) les quatre actions nécessaires à la planification des tâches décrites dans les fichiers **pb1a.pddl** et **pb1b.pddl**. Les fichiers problèmes restent les mêmes, seuls les fichiers domaines changent. Noter les durées des actions: 2, 2, 5, et 10 unités de temps pour respectivement mount, unmount, move et perform. Commenter les plans produits. Comparez les plans obtenus avec les plans de la question 1.

On a plusieurs robots terrestres, mais une zone ne peut comporter qu'un robot à la fois (sauf la base).

**Question 3 (ff):** Modifier dans **dom2.pddl** vos spécifications précédentes (**dom1.pddl**) pour planifier les tâches décrites dans les fichiers **pb2a.pddl** et **pb2b.pddl**. Commenter les plans produits.

**Question 4 (optic):** Modifier dans **dom2d.pddl** vos spécifications précédentes (**dom1d.pddl**) pour planifier les tâches décrites dans les fichiers **pb2a.pddl** et **pb2b.pddl**. **Attention** contrairement à ff le planificateur **optic** ne permet pas les disjonctions dans les conditions des actions. Il faut donc trouver une façon de modéliser l'action Move sans disjonction (les noms proposés pour les actions devraient vous aider). Comparez les plans obtenus avec les plans de la question 2

On ajoute des contraintes de précédence entre tâches, en effet le buttage ne peut se faire qu'après le binage.

**Question 5 (ff):** Modifier dans **dom3.pddl** les spécifications précédentes (**dom2.pddl**) pour planifier les tâches décrites dans les fichiers **pb3a.pddl** et **pb3b.pddl**. Commenter les plans produits

**Question 6 (optic):** Modifier dans **dom3d.pddl** vos spécifications précédentes (**dom2d.pddl**) pour planifier les tâches décrites dans les fichiers **pb3a.pddl** et **pb3b.pddl**. De nouveau, les limitations de d'optic font qu'il faut recourir à une astuce pour exprimer la disjonction sur les tâches indépendantes (e.g. binage) ou pas (e.g. buttage).

**Question 7 :** Quelles sont les critiques que l'on peut faire sur les plans produits par ff, en particulier si on compare les plans avec un robot ou avec plusieurs robots. Est ce qu'optic produit des plans qui ont les mêmes "problèmes".

**Question 8 (optic):** Enfin, afin de rendre le modèle plus réaliste, on considère maintenant que les actions de binage et buttage ont des durées différentes. Modifier dans **dom4d.pddl** le modèle précédant (**dom3d.pddl**) **sans** spécifier d'action perform spécifique, pour planifier les tâches décrites dans les fichiers **pb4a.pddl** et **pb4b.pddl** (Notez dans ces fichiers les déclarations de durée du binage et du buttage).

**Documents à préparer pour l'examen du 26/10 :**

- les plans obtenus avec vos commentaires;
- les sept fichiers dom1.pddl à dom3.pddl, dom1d.pddl à dom4d.pddl que vous aurez complétés.

Le fichier compressé **BE-planif.zip** comporte

- Ce texte du BE (BE Planif 2018-2019.pdf)
- **ewr** : dossier propre à ce BE. Les fichiers domX.pddl et domXd.pddl doivent être complétés, alors que les fichiers pbXa.pddl pbXb.pddl ne doivent pas être modifiés.
  - question1 : fichiers **dom1.pddl**, **pb1a.pddl**, **pb1b.pddl**
  - question2 : fichiers **dom2.pddl**, **pb2a.pddl**, **pb2b.pddl**
  - question3 : fichiers **dom3.pddl**, **pb3a.pddl**, **pb3b.pddl**
  - question5 : fichiers **dom1d.pddl**, **pb1a.pddl**, **pb1b.pddl**
  - question6 : fichiers **dom2d.pddl**, **pb2a.pddl**, **pb2b.pddl**
  - question7 : fichiers **dom3d.pddl**, **pb3a.pddl**, **pb3b.pddl**
  - question8 : fichiers **dom4d.pddl**, **pb4a.pddl**, **pb4b.pddl**

## Annexe

Les deux planificateurs s'utilisent à peu près de la même façon:

Pour les rendre disponibles dans votre environnement, il faut exécuter la commande:

```
module load ff
ou
module load optic/release
```

ff:  
ff s'utilise ainsi:

```
ff -o <fichier-domaine> -f <fichier-problème>
```

```
exemple: ff -o dom3.pddl -f pb3a.pddl
```

optic-clp:  
optic-clp s'utilise ainsi:

```
optic-clp <fichier-domaine> <fichier-problème>
```

```
exemple: optic-clp dom3d.pddl pb3a.pddl
```

Attention, optic-clp par défaut produit une première solution qu'il essaye ensuite d'améliorer. Afin de pouvoir comparer les solutions obtenues, on prendra toujours comme référence le premier plan

produit (même si ce n'est pas le meilleur). Une fois ce premier plan imprimé à l'écran, vous pouvez/devez arrêter le planificateur avec Ctrl-C/Ctr-C.

Si vous désirez sauver le résultat d'une commande dans un fichier (tout en le voyant à l'écran), vous pouvez ajouter

```
| tee <fichier-resultat>
```

à la fin de votre commande. Exemple

```
optic-clp dom3d.pddl pb3a.pddl | tee res3da.txt
```

sauvera le résultat de la commande "optic-clp dom3d.pddl pb3a.pddl" dans "res3da.txt", tout en affichant ce résultat à l'écran.